Continuous Security in Modernen Webanwendungen

Sicherheit eingebaut

Martin Reinhardt (Holisticon AG)

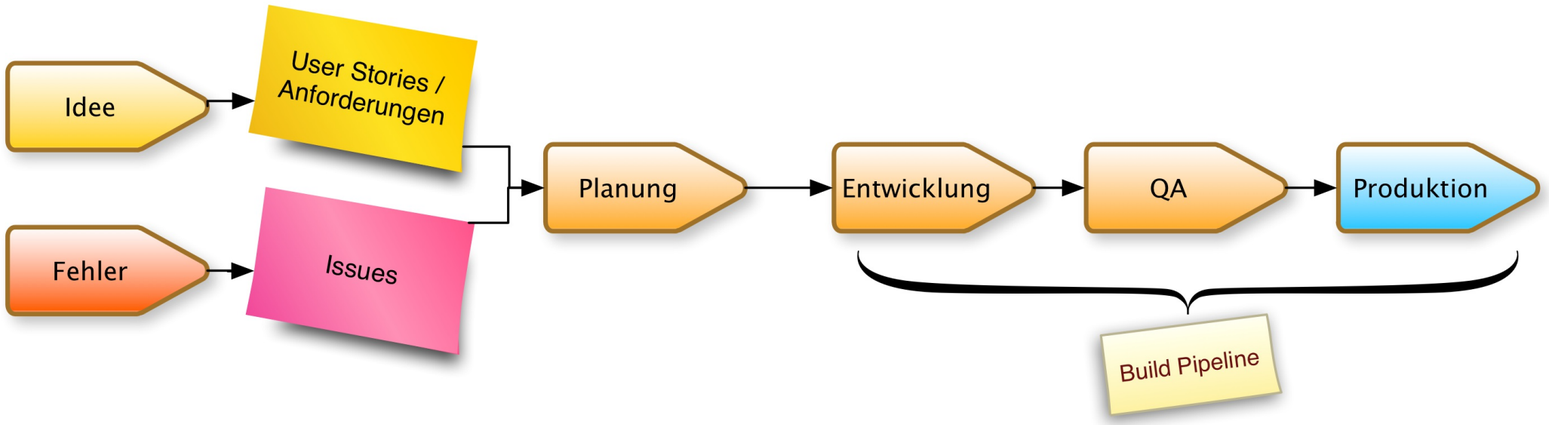
@mreinhardt

Agenda

- Continuous Delivery
- IT-Sicherheit
  - Methodik
  - Agil & Security
- Continuous Security
  - Automatisierung
  - Tools
- Ausblick
- Links

Continuous Delivery

- logische Fortsetzung von Continuous Integration



- Die Idee dahinter
  - Software mit agilen Methoden kann nicht komplett (manuell) getestet werden
  - Alle 2 Wochen gesamten Funktionsumfang abtesten ist utopisch
  - Fast Feedback durch schnelle Releases
  - 15 min from patch to prod

Warum Build Pipelines

"One of the challenges of an automated build and test environment is you want your build to be fast, so that you can get fast feedback, but comprehensive tests take a long time to run. A deployment pipeline is a way to deal with this by breaking up your build into stages. Each stage provides increasing confidence, usually at the cost of extra time. Early stages can find most problems yielding faster feedback, while later stages provide slower and more through probing. Deployment pipelines are a central part of Continuous Delivery"

Martin Fowler

- Agile Softwareentwicklung arbeitet kleinteilig
  - Software oft und zuverlässig in Produktion
  - Nutzung der IDE != Automatisierung
  - Wesentlich ist dabei die Build Pipeline
- Wie?
  - Geschwindigkeit
  - Automatisierung

Build Pipeline

- wesentliches Ziel ist schnelles Feedback, also Geschwindigkeit
  - Einzelne Schritt schnell abarbeiten (5 - 10 Minuten)
  - Möglichst früh Fehler finden (**Unit-Tests**)
- unabhängig vom (Build-)Tool
  - Build muss reproduzierbar sein
- CI-Tools
  - Jenkins, Go, Travis CI, XCode Bots ...
  - TestTools
- Quellcode muss Continuous Delivery gerecht werden
- Warum nicht auch für Security?

#n IT-Sicherheit

holisticon

Warum das Ganze?

- NSA
- BDSG
- DSGVO
- Kosten
- Exploits
  - CVE-2016-5000 - Apache POI Information Disclosure via External Entity Expansion (XXE)
  - CVE-2016-4216 - Adobe XMP Toolkit for Java Information Disclosure via External Entity Expansion (XXE)
  - CVE-2016-3081 - Remote code execution vulnerability in Apache Struts when dynamic method invocation is enabled
  - CVE-2015-8103 - Remote code execution vulnerability in Jenkins remoting; related to the Apache commons-collections

Black Duck - Open Source Security Analysis

- Stand von Open Source Security in kommerziellen Anwendungen bit.ly/2yfsD2x
  - 95% der Anwendungen enthalten OSS
  - 67% der Anwendungen enhalten OSS Schwachstellen
  - Durchschnittsalter von bekannten Schwachstellen in OSS: 1894 Tage

holisticon

OWASP Top 10

- Kritischsten Risiken in Webanwendungen
- A9 - Nutzung von Komponenten mit bekannten Schwachstellen
- Schwer zu erkennen
- Bewusstsein auf Entwicklungsseite
- Sichtbarkeit
- Toolunterstützung nicht vorhanden
- Patching erfordert erneut Codeänderungen

Arten von Tests

- Funktionale Sicherheitstests
- Schwachstellen-Scanning / Fuzzing
- Penetrationstests

Continuous Security

- Testen ist nicht alles, bei Entwicklung auch nicht
- Warum also nicht auch bei Security?
- logischer Schritt für Automatisierung
- Security muss auf verschieden Ebenen betrachtet werden
  - Code & Architektur (Sonar)
  - Integrations-Tests (bdd-security, zap, owasp)

Security ganzheitlich

- Thema einarbeiten
  - ☐ Erstellung von Security Guidelines
  - ☐ Berücksichtigung von Sicherheit im Rahmen der Erstellung von User Stories
  - ☐ Codescans durchführen
  - ☐ Peer-Reviews durch einen Security Champion durchführen
  - ☐ Penetrationstests einplanen
- Planing
- Secure Scrum
- Thread Model
- Analog zur restlichen Softwareentwicklung

Secure Scrum

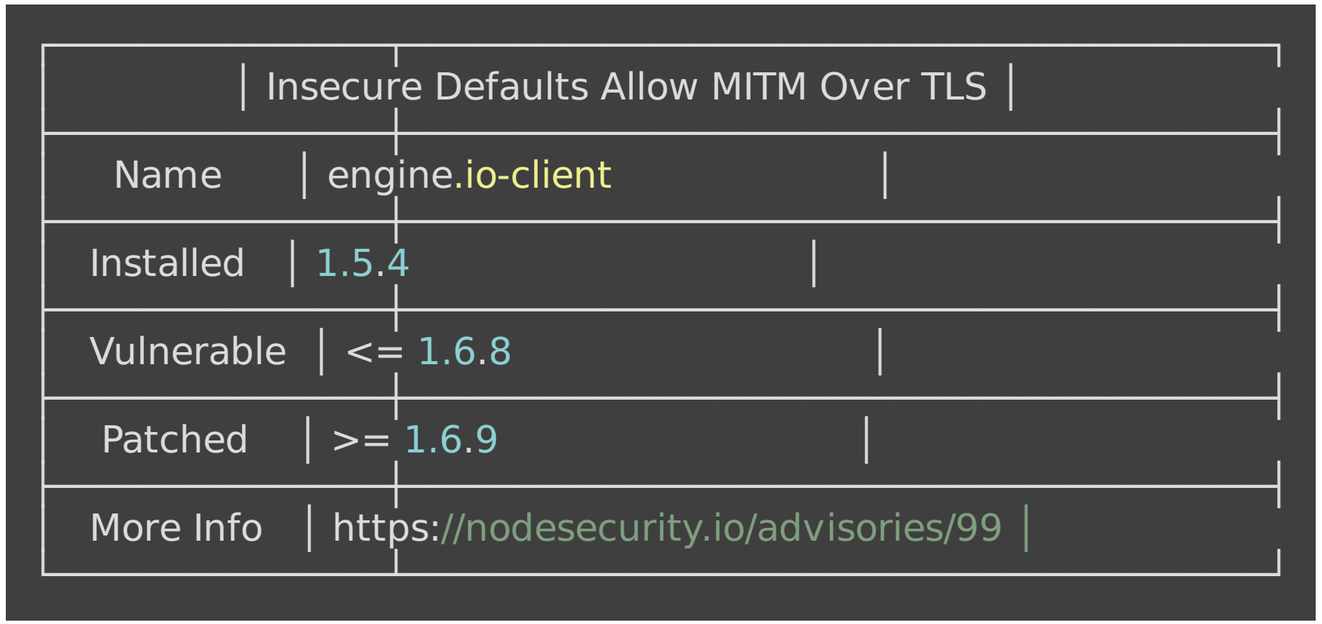
- Setzt auf eigentliche Scrum-Prozess auf
- Security relevante User-Stories im Team identifizieren
- Jede User-Story mit Loss-Value versehe und mit S-Marks gekennzeichnet
- S-Marks verweisen auf S-Tag
- beschreibt das entsprechende Security-Problem als auch mögliche Lösungsansätze
- Wird User Story mit einem S-Mark implementiert, muss im selben Sprint der entsprechende S-Tag implementiert werden
- Alternativ: Security Backlog im regulären Sprint

Continuous Security Testing

- Tools mittlerweile verfügbar
- meist setzen diese auf OWASP auf
- Integration nicht schwieriger als bei DevOps

Node Security Project (NSP)

- Prüfung der Abhängigkeiten auf bekannte Schwachstellen
- **Ein Paket** zu installieren
- Schlägt korrigierte Version vor



holisticon

OWASP dependency-check

- Seit 2012 verfügbar WASP Top 10 2013 entry: A9 - Using Components with Known Vulnerabilities
- Verfügbar in verschiedenen Varianten: Ant, Maven, Gradle, SBT, Jenkins
- Analyse der Abhängigkeiten zu bekannten Schwachstellen für Java & .NET
- Experimentielle Unterstützung
  - CocoaPods
  - Swift Package Manager
  - Python
  - PHP (composer)
  - Node.js
  - Ruby

- Prinzipiell kann jede gefundenen Schwachstelle zu Buildfehler führen

```
[ERROR] Failed to execute goal org.owasp:dependency-check-maven:1.4.0:check (default) ...
[ERROR]
[ERROR] Dependency-Check Failure:
[ERROR] One or more dependencies were identified with vulnerabilities
        that have a CVSS score greater then '5.0':
[ERROR] commons-httpclient-3.1.jar: CVE-2014-3577
[ERROR] mysql-connector-java-5.1.37.jar: CVE-2014-0001, CVE-2013-2378,  ....
[ERROR] tomcat-embed-core-8.0.33.jar: CVE-2016-3092, CVE-2013-2185, CVE-2002-0493
```

- Nicht praktikabel, deswegen sind Ausnhamen nötig
  - Kann reduzierter Common Vulnerability Scoring System-Score (CVSS) gewählt werden
  - Ausnahmen festlegen

HTML - Report

DEPENDENCY-CHECK

Project: angular-spring-boot-webapp

Scan Information (show all):

- *dependency-check version*: 1.4.0
- *Report Generated On*: Oct 1, 2016 at 08:24:04 CEST
- *Dependencies Scanned*: 128
- *Vulnerable Dependencies*: 3
- *Vulnerabilities Found*: 106
- *Vulnerabilities Suppressed*: 0
- ...

Display: Showing Vulnerable Dependencies (click to show all)

Dependency	CPE	GAV	Highest Severity	CVE Count	CPE Confidence	Evidence Count
commons-httpclient-3.1.jar	cpe:/a:apache:commons-httpclient:3.1 cpe:/a:apache:httpclient:3.1	commons-httpclient:commons-httpclient:3.1	Medium	3	LOW	17
mysql-connector-java-5.1.37.jar	cpe:/a:mysql:mysql:5.1.37	mysql:mysql-connector-java:5.1.37	High	98	HIGHEST	22
tomcat-embed-core-8.0.33.jar	cpe:/a:apache:tomcat:8.0.33	org.apache.tomcat.embed:tomcat-embed-core:8.0.33	High	5	HIGHEST	16

Mit festgelegten Ausnahmen

Dependency-Check is an open source tool performing a best effort analysis of 3rd party dependencies; false positives and false negatives may exist in the OWASP be held liable for any damages whatsoever arising out of or in connection with the use of this tool, the analysis performed, or the resulting report.

Project: angular-spring-boot-webapp

Scan Information (show all):

- *dependency-check version*: 1.4.0
- *Report Generated On*: Oct 1, 2016 at 08:47:00 CEST
- *Dependencies Scanned*: 102
- *Vulnerable Dependencies*: 0
- *Vulnerabilities Found*: 0
- *Vulnerabilities Suppressed*: 0
- ...

Display: Showing Vulnerable Dependencies (click to show all)

Dependency	CPE	GAV	Highest Severity	CVE Count	CPE Confidence	Evidence Count

Dependencies

This report contains data retrieved from the National Vulnerability Database.

- Ausnahmen werden in eigenem XML-Format festgelegt

```
<suppress>
   <notes>
      <![CDATA[This suppresses false positives identified on spring security. ]]>
   </notes>
   <gav regex="true">org\.springframework\.security:spring.*</gav>
   <cpe>cpe:/a:mod_security:mod_security</cpe>
   <cpe>cpe:/a:springsource:spring_framework</cpe>
   <cpe>cpe:/a:vmware:springsource_spring_framework</cpe>
</suppress>
```

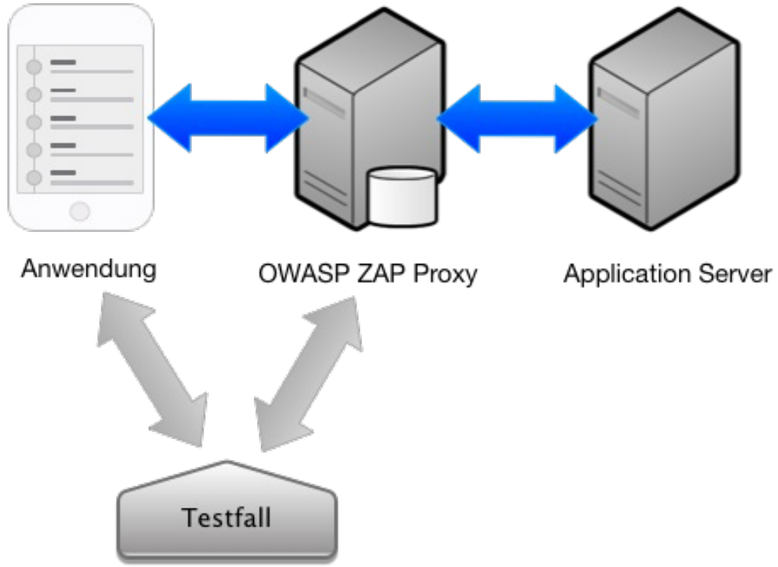
OWASP Zed Attack Proxy (ZAP)

Features

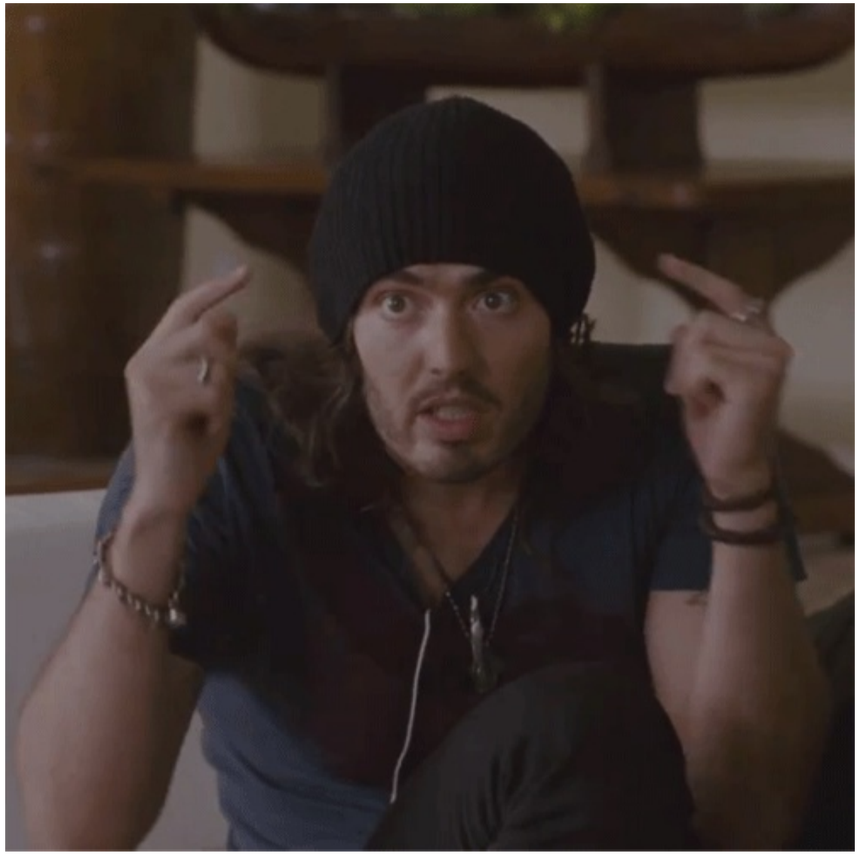
- Intercepting Proxy
- Automated Scanner
- Passive Scanner
- Brute Force Scanner
- Fuzzer
- Port Scanner
- Spider
- Web Sockets
- REST API Scanning (OpenAPI/Swagger)

Funktionsweise

- Installation auf separaten Umgebung

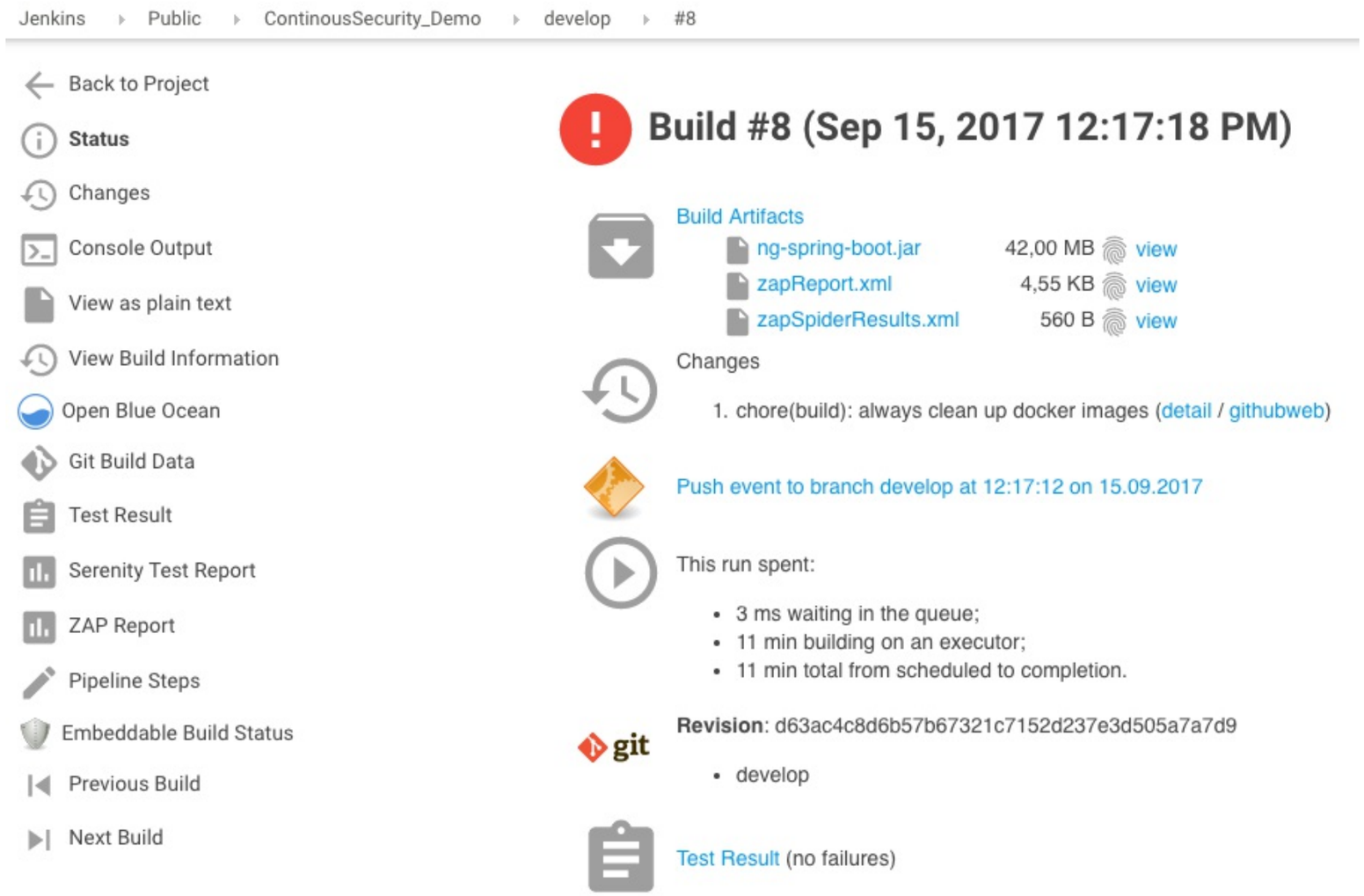


- Scan der Anwendung
- Proxy während Testausführung



Integration in Pipeline

- Maven Plugin: github.com/hypery2k/zap-maven-plugin



holisticon

ZAP Scanning Report

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	0
Low	2
Informational	0

Alert Detail

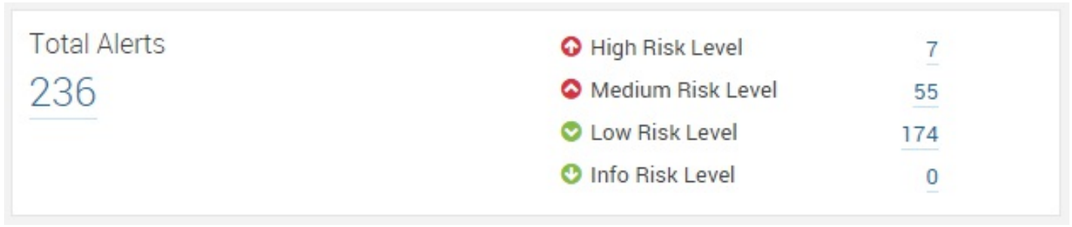
Low (Medium)	X-Content-Type-Options Header Missing
Description	The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.
URL	http://ngspring:41180/scripts/scripts.d13bbe30.js
Method	GET
Parameter	X-Content-Type-Options
URL	http://ngspring:41180/styles/vendor.0549f159.css
Method	GET
Parameter	X-Content-Type-Options
URL	http://ngspring:41180/styles/main.be748ac2.css

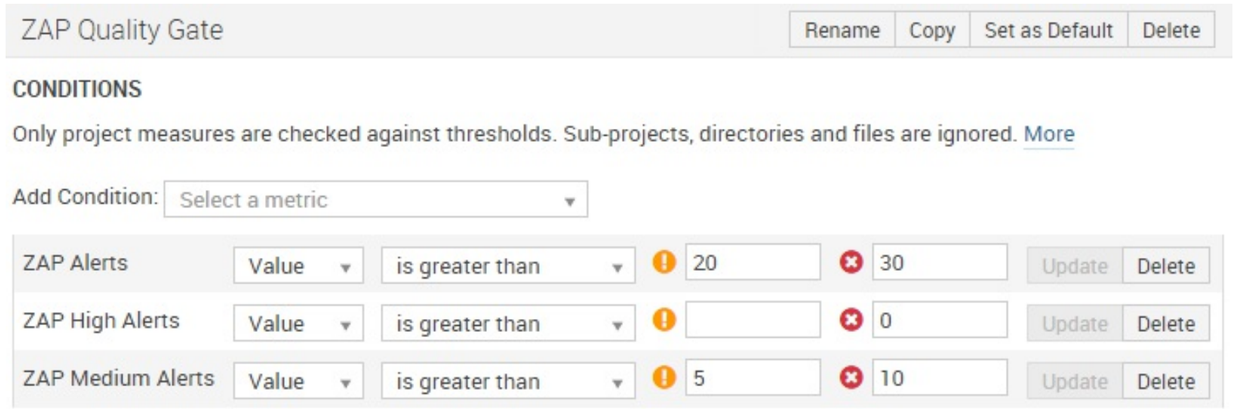
Integration in Build

- Einfache Integration
- Erweiterung möglich (Selenium Tests)

```
<plugin>
    <groupId>de.martinreinhardt-online</groupId>
    <artifactId>zap-maven-plugin</artifactId>
    <configuration>
        <zapHost>localhost</zapHost>
        <zapPort>44444</zapPort>
        <failingRiskCodeThreshold>5</failingRiskCodeThreshold>
        <targetUrl>http://ngspring:41180/</targetUrl>
        <authenticationType>form</authenticationType>
        <username>user</username>
        <password>password</password>
        <shouldRunAjaxSpider>true</shouldRunAjaxSpider>
    </configuration>
    ...
</plugin>
```

Sonar Integration





github.com/pdsoftplan/sonar-zap

holisticon

AWS absichern

- Security Monkey
  github.com/Netflix/security_monkey
- Monitoring für Sicherheitsprobleme
- Für große verteilte AWS-Anwendungen

Fazit & Ausblick

- Bibliotheken = Sicherheitsrisiko
  - gerade im modernen Umfeld
  - zeitnahe Aktualisierung nötig
  - automatisierbar
- Absicherung möglich
  - Penetrationstests durch DevOps einfach automatisierbar
  - viele Tools im Bereich Testing & Automatisierung
  - Spring Vault projects.spring.io/spring-vault/
- Feedback ist ein Muss
- unerlässlich ein Sicherheitsbewusstsein im Team aufzubauen

"There is no one-size-fits-all solution to the complex problem of implementing a deployment pipeline."

Continuous Delivery, J. Humble, D. Farley

*"**There are only two types of companies**: those that have been hacked, and those that will be"*

Robert Miller, FBI Director, 2012

Links

- Beispiel Anwendung
- OWASP Top 10
- OWASP Cheat Sheet
- BSI Empfehlungen zu Webanwendungen
- DevOps – Testautomation I – Infrastructure as Code
- Rock CI mit Jenkins 2 und Docker
- Beispiel Serenity
- ZAP Blog
- Your code as a crime scene
- Secure Scrum

About me

- Martin Reinhardt (Holisticon AG)

- github.com/hypery2k
- twitter.com/mreinhardt

holisticon

Präsentation

holisticon.github.io/presentations/isd-continuous-security/

holisticon